

On page 4 of the Answer, the Examiner alleges as follows:

Kekic teaches the affirmative step of abstracting interface data through the use of Kekic's element manager. Kekic teaches an element manager that "is an abstract representation of the managed computer network element server that when executed on manager 404 of managed element server 314 manages and monitors the managed computer network element associated with element manager 800" (Kekic, column 26, lines 31-58). This element manager functions by first interrogating network elements using a discovery engine (Kekic, column 7, lines 20-26). When the discovery engine finds an appropriate network device, using abstraction it creates a "managed element object" that represents the physical device (Kekic, column 7, lines 27-33). Alternatively, a user can enter data manually, and the system will "build" an element manager by abstracting the data (Kekic, column 9, lines 1-8). The element manager forms an abstraction representation of attributes such as ports (i.e., hardware interfaces) and any applications (i.e., software interfaces) on the network device, among other requirements (Kekic, column 24, lines 17-21 and column 25, lines 60-65). Additionally, Kekic teaches an element manager running as a JAVA module (Kekic, column 8, lines 48-60). From the described functionality, Kekic clearly teaches the affirmative step of abstracting both hardware and software interfaces.

Appellant replies to the Examiner's Answer by addressing each of the above allegations:

1) ***"Kekic teaches the affirmative step of abstracting interface data through the use of Kekic's element manager."***

REPLY: Applicant respectfully submits that Kekic -- as evidenced by the excerpts relied upon by the Examiner -- does not teach or suggest the affirmative step of "abstracting

interface" using a processor, as expressly set forth in claims and 1 and 11. In fact, the element manager 800 referenced by the Examiner comprises an abstract representation managed computer network element, not any feature or functionality that actually performs the step of "abstracting interface data." See, e.g., Kekic, Col. 26, lines 41-58 (emphasis added).

Further, Appellant respectfully submits that Kekic is a voluminous reference that discloses numerous features and functionality. In composing the rejection, the Examiner merely cobbles together various excerpts from Kekic that are alleged to disclose the claimed functionality of "abstracting interface data" using a processor. As set forth below, however, the cited excerpts fail to disclose any feature or functionality that specifically "abstracts interface data" in the manner claimed.

2) ***"Kekic teaches an element manager that "is an abstract representation of the managed computer network element server that when executed on manager 404 of managed element server 314 manages and monitors the managed computer network element associated with element manager 800" (Kekic, column 26, lines 31-58)."***

REPLY: Appellant agrees that Kekic discloses an element manager 800 that is an abstract representation of the managed

computer network element server that when executed on manager 404 of managed element server 314 **manages and monitors** the managed computer network element associated with element manager 800. However, Applicant respectfully submits that while element manager 800 performs the affirmative steps of managing and monitoring, it does not perform the affirmative step of "abstracting interface data" using a processor as required by the claims. Appellant further submits that just because element manager 800 is itself an abstract representation, does not mean that it performs the active step of "abstracting interface data."

3) ***"This element manager functions by first interrogating network elements using a discovery engine (Kekic, column 7, lines 20-26)."***

REPLY: Appellant respectfully submits that the discovery engine does not "abstract interface data" as required by the pending claims. The cited excerpt provides:

The manager of this invention includes a discovery engine that automatically interrogates each host in the computer network ***to determine whether the host is running a network management agent***. Upon detection of a management-enabled computer network element, the discovery engine attempts to associate one of the plurality of element managers with the computer network element.

Col. 7, lines 20-26 (emphasis added).

Rather, Appellant respectfully submits that the discovery engine merely interrogates hosts to determine whether a host is running a network management agent. Appellant respectfully submits that this is not the same as "abstracting interface data" using a processor. Further, Appellant respectfully submits that it is not readily clear that the "manager" and "discovery engine" referenced in the cited excerpt refers to (or comprises) the "element manager" discussed above.

4) *"When the discovery engine finds an appropriate network device, using abstraction it creates a "managed element object" that represents the physical device (Kekic, column 7, lines 27-33)."*

REPLY: Appellant respectfully submits that the discovery engine does not "abstract interface data" as required by the pending claims. The cited excerpt provides as follows:

If the discovery engine is able to associate the computer network element with one with one of the plurality of element managers, the discovery engine calls a process that uses the element manager to create a managed element object, and creates a poll server and an event engine for the managed computer network element.

Col. 7, lines 27-33.

Rather, Appellant respectfully submits that if the discovery engine is able to associate a computer network element

with one with one of the plurality of element managers, then the discovery engine calls a process that uses the element manager to create a managed element object. Appellant respectfully submits that this does not amount to abstracting interface data.

5) *"Alternatively, a user can enter data manually, and the system will "build" an element manager by abstracting the data (Kekic, column 9, lines 1-8)."*

REPLY: Appellant respectfully submits that having a user "manually" build an element manager is not the same as "abstracting interface data" using a processor.

6) *"The element manager forms an abstraction representation of attributes such as ports (i.e., hardware interfaces) and any applications (i.e., software interfaces) on the network device, among other requirements (Kekic, column 24, lines 17-21 and column 25, lines 60-65)."*

REPLY: Appellant respectfully submits that the excerpts cited by the Examiner do not teach or suggest any affirmative "abstraction" step that is performed by the element manager, or, for that matter, any other feature or functionality disclosed by Kekic. Rather, the excerpts cited merely disclose computer network element attribute types and typical managed elements, but do not teach or suggest any affirmative "abstraction" step:

Managed element server 314 supports the following computer network element attribute types:

(i) element hotspots representing (a) components, e.g., ports, LEDs, jacks, meters, connectors, switches, etc., (b) action buttons, and (c) embedded graphs. . . .

Kekic, Col. 24, lines 17-21.

Managed elements 701 are SNMP-enabled devices which are managed by server 314. Typical managed elements 701 include hubs, routers, switches, bridges, and work stations. In general, a managed element can be any hardware or software entity that implements SNMP protocol, e.g., printers, applications, etc.

Kekic, Col. 25, lines 60-65.

7) *"Additionally, Kekic teaches an element manager running as a JAVA module (Kekic, column 8, lines 48-60)."*

REPLY-Applicant respectfully disagrees that Kekic discloses an element manager running as a JAVA module. Rather, Appellant submits that Kekic merely discloses a "managed element server client" that is implemented as JAVA module:

The managed element server also interacts with managed element server clients. In one embodiment, the **managed element server client** is implemented as a JAVA applet and is running inside a World-Wide Web Browser or a JAVA Applet Viewer. The JAVA applet is downloaded from the managed element server. The JAVA applet includes information that allows a user a) to monitor the operation of each of the managed computer network elements, b) to edit the event management for the managed computer network elements by reconfiguring the event management model in the element manager object for the network element, and c) in one embodiment, to

use a visual element manager builder that permits both building and editing of element managers.

Kekic, Col. 8, lines 48-60 (emphasis added).

Further, Appellant respectfully submits that Kekic does not teach or suggest any JAVA module that performs the affirmative step of "abstracting interface data" as recited by the pending claims.

8) *"From the described functionality, Kekic clearly teaches the affirmative step of abstracting both hardware and software interfaces."*

REPLY: Appellant respectfully submits that, for at least the reasons set forth above, Kekic does not teach or suggest the affirmative step of "abstracting interface data" as set forth in the pending claims.

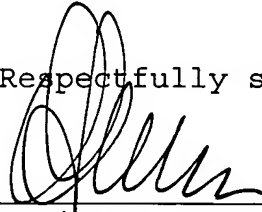
CONCLUSION

In view of the foregoing, it is respectfully submitted that the present application is in condition for allowance, and an early indication of the same is courteously solicited. The Examiner is respectfully requested to contact the undersigned by telephone at the below listed telephone number, in order to expedite resolution of any issues and to expedite passage of the present application to issue, if any comments, questions, or suggestions arise in connection with the present application.

To the extent necessary, a petition for an extension of time under 37 CFR § 1.136 is hereby made.

Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account No. 50-0206, and please credit any excess fees to the same deposit account.

Respectfully submitted,



Ozzie A. Farres

Registration No. 43,606

Hunton & Williams LLP
1900 K Street, NW
Washington, D.C. 20006-1109
Tel. (202) 955-1500
Fax (202) 778-2201

Date: July 5, 2006